



프로젝트 계획서



프로젝트 주제

- HR Bank



프로젝트 구성원과 R&R

팀원	주요 담당
공한나	직원 정보 수정 이력 관리 기능 및 공통 모듈 개발
김현호	데이터 백업 관리 기능 및 파일 관리 기능 개발
도효림	직원 정보 관리 기능 개발
양성준	직원 통계 조회 기능 및 대시보드 개발
최규원	부서 관리 기능 개발



프로젝트 일정 요약

항목	기간	내용
기획 및 요구사항 정리	4/21(월)	프로젝트 방향성 논의, 기능 리스트 작성, ERD & 엔티티 구조 설계, 프로젝트 세팅
1차 개발 스프린트	4/22 ~ 4/26	주요 기능 개발
중간 점검 & 회고	4/26(토)	기능 시연 + 피드백 수렴
2차 개발 스프린트	4/26 ~ 4/28	세부 기능 개발, 테스트, 버그 수정
개발 완료	4/28(월)	개발 완료(오전), 발표 자료 작성(오후)
최종 발표 & 회고	4/29(화)	발표 자료 정리 및 시연



프로젝트 세부 계획



기술 스택 및 협업 도구

분류	사용 예정 도구
Backend	Spring Boot, Spring Data JPA, QueryDSL, Redis, MapStruct
Database	PostgreSQL
API 문서화	Swagger

협업 도구	Discord, GitHub, Notion
일정 관리	GitHub Issues
배포	Railway.io

규칙 수립

항목	내용
네이밍 컨벤션	camelCase (변수, 함수), PascalCase (클래스), kebab-case (파일), com.team6.hrbank+소문자 (패키지명), 소문자+복수형+snake_case (테이블명), 소문자+snake_case (컬럼명)
커밋 컨벤션	feat : 새로운 기능 추가, fix : 버그 수정, refactor : 코드 리팩토링, docs : 문서 수정, style : 코드 포매팅, 세미콜론 누락, 코드 변경이 없는 경우, test : 테스트, chore : 기타 변경사항(https://chlolisher.tistory.com/173) ex) feat: XX 기능 추가
브랜치 전략	GitHub Flow 전략: main , dev , feature/기능명 , bugfix/기능명
PR 규칙	전원 Approve 시 Merge + review할 사항 있으면 각자 Review
예외 처리	enum 파일로 예외 코드 관리+GlobalExceptionHandler로 전역 예외 처리

협업 규칙

항목	시간	내용
데일리 스크럼	오전 9시(1교시), 오후 2시	작업 계획 요약하여 문서화(오전), 문제 상황 문서화(오후)

항목	설명	예시
이슈 관리	[이름 이슈 생성 날짜] 작업 내용 요약	[공한나 0421] 직원 등록 기능 개발
서브 이슈 관리	commit 단위의 작업 내용	검증 로직 구현
커밋 단위	기능 및 메서드 단위로 커밋	직원 등록 기능 구현
PR 단위	1차: 기본 기능 완성, 2차: 세부 로직 보완, 3차: 버그 수정	1차: 직원 관리 기능 완성
브랜치 형식	feature/티켓번호-작업내용	feature/10-직원-등록

산출물

항목	내용
계획서	
백로그	프로젝트에서 구현해야 할 기능 목록
ERD	
발표 자료	
시연 영상	

GitHub 리포지토리	README 작성
백엔드 배포 링크	프로젝트 종료일 기준 2주 동안 사이트 active 상태 유지 필수
4L 회고 문서	1. Loved – 이번 스프린트에서 좋았던 점 2. Learned – 이번 스프린트를 통해 새롭게 배운 점 3. Loathed – 아쉬웠거나 힘들었던 점 4. Longed for – 더 진행해보고 싶은 요소나 필요했던 지원
GitHub 팀문서	
팀원별 개발 리포트	1. 프로젝트 개요 2. 담당한 작업 3. 기술적 성과 4. 문제점 및 해결 과정 5. 협업 및 피드백 6. 코드 품질 및 최적화 7. 향후 개선 사항 및 제안

백로그(티켓)

<직원 정보 관리>

	기능 목록	설명	비고	담당자
1.	직원 등록	{이름}, {이메일}, {부서}, {직함}, {입사일}, {프로필 이미지}를 통해 직원을 등록한다.	사원번호는 자동으로 부여한다. 상태는 자동으로 재직 중 상태로 초기화 한다. 이메일 중복 체크를 한다.(유효성 검사) 선택적으로 {프로필 이미지}를 업로드 할 수 있다.	도효림
2.	직원 정보 수정	사용자는 {사원번호}를 제외한 다른 속성을 모두 수정할 수 있다.	이메일 중복 불가, 유효성 검사	도효림
3.	직원 삭제	사용자는 직원을 삭제할 수 있고, 프로필 이미지도 같이 삭제 된다.	퇴사는 삭제가 아닌 수정으로 처리한다.	도효림
4.	직원 목록 조회	{이름 또는 이메일}, {부서}, {직함}, {입사일}, {상태}로 직원 목록을 조회할 수 있다.	조건이 여러 개인 경우 모든 조건을 만족한 결과로 조회	도효림

			<p>{입사일} 범위로 직원 목록을 조회할 수 있다. → 날짜 범위 조건</p> <p>{상태} (예 재직중, 휴직 중, 퇴사) 로 직원 목록을 조회할 수 있다. →완전 일치 조건</p> <p>[정렬] - 1개만 선택적으로 가능하다</p> <p>{이름}, {입사일}, {사원번호}로 정렬할 수 있다. [페이지네이션]</p> <p>{이전 페이지의 마지막 요소 ID} 를 활용해 페이지네이션 기반으로 목록을 조회할 수 있다.</p>	
5.	직원 정보 상세 조회	사용자는 {사원번호} 로 상세정보를 조회할 수 있다.		도효림

<부서 관리>

	기능 목록	설명	비고	담당자
1	부서 등록	{이름}, {설명}, {설립일} 을 입력해 부서를 등록	{이름} 은 중복 불가	최규원
2	부서 수정	기존 부서의 {이름}, {설명}, {설립일} 을 수정	{이름} 은 중복 불가	최규원
3	부서 삭제	부서를 삭제	{소속된 직원} 이 없는 경우에만 부서 삭제 가능	최규원
4	부서 상세 조회	{부서 ID} 로 상세 정보 조회		최규원

5	부서 목록 조회	{이름 또는 설명}으로 부서 목록 조회	{이름 또는 설명} 기준 으로 조회 - 커서 기반 페이지네이션: {이전 페이지의 마지막 요소 ID}를 활용 - 정렬: {부서명 또는 설립일} 기준으로 1개의 정렬 조 건을 가짐	최규원
---	----------	--------------------------	--	-----

<직원 정보 수정 이력 관리>

	기능 목록	설명	비고	담당자
1	이력 등록 관련 이벤 트 발행 및 리스너	직원 정보 등록/수정 시 이력 자동 등록을 위한 이벤트 기반 처리 로직 개발		공한나
2	이력 등록 API	직원 추가, 직원 정보 수정 화면에서 입력하 는 메모 정보를 통해 이력 등록	- 직원 정보 수정 시 변경된 필드와 기존 값을 비교하는 로직 개발 - {IP 주소} 추출 후 저 장하는 유틸리티 메 서드 개발	공한나
3	이력 목록 조회 API	{대상 직원 사번}, {메 모}, {IP 주소}, {시 간}, {유형} 조건 적용 하여 이력 목록 조회	- {변경 상세 내용}은 포함하지 않음 - {IP 주소}, {시간}으 로 정렬 구현 - 커서 기반 페이지네 이션 구현	공한나
4	이력 상세 변경 내용 조회 API	{id}로 {변경 상세 내 용} 단건 조회		공한나

<데이터 백업 및 파일 관리>

	기능 목록	설명	비고	담당자
1	파일 메타 정보 저 장	파일의 메타 정보와 실 제 파일을 저장	- 메타 정보는 DB에 저장 - 실제 파일은 로컬 디스크에 저장	김현호

2	백업 파일 다운로드	메타 정보 ID 기반으로 실제 파일을 찾아 다운 로드	- 실제 파일을 메타 정보를 통해 다운로 드	김현호
3	백업 생성	직원 데이터 변경 여부 판단 후 백업 진행	- 데이터 백업 프로세 스 1시간마다 자동 반 복 - 변경이 없을 경우 백업 건너뛸 - CSV파일로 디스크 에 저장 - {작업자}는 요청자 의 IP 주소 - 백업 실패시 파일 삭제 및 에러 로그 .log 파일에 저장	김현호
4	가장 최근 백업 정 보 조회	가장 최근 시각의 백업 이력 조회	- {상태} 완료 조건 필수	김현호
5	백업 목록 조회	{작업자}, {시작 시간}, {상태} 조건 기반 이력 목록 조회	- 복합 조건 필요 - 조건 1개 기준으로 정렬 - Cursor 기반 페이 지네이션	김현호

<직원 통계/집계 및 대시보드 관리>

	기능 목록	설명	비고	담당자
1.	총 직원 수 집계	모든 부서의 모든 직 원 수를 집계	- 실시간 조회 → 성 능측정 → 배치/캐싱 전환	양성준
2.	최근 일주일 수정 이 력 건수 집계	최근 7일 동안의 수정 이력 수를 집계	- 실시간 조회 → 성 능측정 → 배치/캐싱 전환	양성준
3.	이번 달 입사자 수 집 계	이번 달에 입사한 사 람 수를 월 단위로 집 계	- 저번 달 직원 수와 이번 달 직원 수의 차 이로 집계 - 실시간 조회 → 성 능측정 → 필요시 배 치/캐싱 전환 - 추이 분석을 위해 DB 저장, 조회 시 캐 시로 응답	양성준
4.	마지막 백업 시간 조 회	마지막 백업이 언제 수행됐는지 조회	- 백업 이력에서 가장 최근 완료된 배치 작 업 시간을 기준으로	양성준

			함 - 실시간 조회 → 성능측정 → 필요시 배치/캐싱 전환	
5.	직원 수 변동 추이 통계 (일별, 주별, 월별, 분기별, 연도별)	해당 기간 별 직원 수 변동 추이 통계	-일 별 직원 수 추이를 바탕으로 집계하여 월 별 직원 수 변동 추이 통계 - 실시간 조회 → 성능측정 → 필요시 배치/캐싱 전환 - 추이 분석을 위해 DB 저장, 조회 시 캐시로 응답	양성준
6.	부서 별 직원 분포 통계	부서별 직원 분포를 매일 집계하여 통계	- 각 부서에 몇 명이 소속되어 있는지, 비율은 얼마나 되는지, 부서에 몇 명이 들어왔고 몇 명이 나갔는지 집계하여 통계 - 실시간 조회 → 성능측정 → 필요시 배치/캐싱 전환 - 추이 분석을 위해 DB 저장, 조회 시 캐시로 응답	양성준
7.	직무 별 직원 분포 통계	직무별 직원 분포를 매일 집계하여 통계	- 각 직무의 직원 비율은 얼마나 되는지, 직무에 몇 명이 들어왔고 몇 명이 나갔는지 집계하여 통계 - 실시간 조회 → 성능측정 → 필요시 배치/캐싱 전환 - 추이 분석을 위해 DB 저장, 조회 시 캐시로 응답	양성준

<공통 모듈 관리>

	기능 목록	설명	비고	담당자
1	공통 Exception 핸들러 클래스 개발	GlobalExceptionHandler 클래스로 Exception 전역 처리		공한나

2	에러 코드 정의	Enum으로 에러 유형별 에러 코드 정의		공한나
3	에러 응답 정의	에러 응답의 표준 포맷 정의		공한나

예상 문제점 / 아쉬운 점

- 구체적인 문서화의 필요성
- 팀원 간 개발 일정 및 커밋 단위 관리할 때의 어려움
- 팀원들과의 효과적인 소통 방법 마련의 필요성

개선 사항

- 데일리 스크럼 진행할 때, 구체적으로 문서화하기 (한 일 - 할 일 - 공유사항 및 특이사항 형식)
- 이슈 단위와 커밋 단위를 명확하게 정하기 (API, 메서드 단위)
- 하루 2회 정기적으로 소통하며 각자의 개발 진행 상황과 일정을 공유하기